Keystone: An Open Framework for Architecting Trusted Execution Environments

Dayeol Lee, David Kohlbrenner, Shweta Shinde, Krste Asanovic, Dawn Song

Dept. of Electrical Engineering and Computer Sciences
University of California, Berkeley
Trusted Execution Environments (TEEs)

- Trusted Execution Environments (TEEs)
- Keystone: an Open Framework for Architecting Trusted Execution Environments
- OS / Hypervisor
- RAM
- Sensitive App
- Other Apps
- Ring 3
- Ring 0 - 2
- OS / Hypervisor
- RAM
- Trustworthy Hardware
- Protected Memory “Enclave”
- Confidentiality
- Integrity
- Remote Attestation

Trusted
Untrusted

Ring 3
Ring 0 - 2
Trustworthy Hardware
Protected Memory “Enclave”
Confidentiality
Integrity
Remote Attestation
Trusted Execution Environments (TEEs)

Reducing Trusted Computing Base (TCB)
Challenges in Existing TEEs

Security

Performance

Functionality

SGX

TrustZone

SEV
Challenges in Existing TEEs

- Security
- Performance
- Functionality

Fixed Design Decisions

- Intel SGX
- ARM TrustZone
- AMD SEV
Challenges in Existing TEEs

Closed-Source Hardware

Security

Fixed Design Decisions

Performance

Functionality
Technical Contributions

Keystone: Customizable RISC-V TEEs

- Fine-Grained Configuration
- Modular Extensions
- Minimal TCB
- No µarch Modification

Framework

- Extensive benchmarking
- Real-world applications
- Multi-platform deployment

Open-Source

- Full-stack available
- Community-driven efforts
- TEE verification & research
Keystone Architecture and Trust Model

Keystone: an Open Framework for Architecting Trusted Execution Environments

- User (U-mode)
- Supervisor (S-mode)
- Machine (M-mode)
- Trusted Hardware

Optional HW

Root of Trust

Trusted Hardware

RISC-V

C0 C1 C2 C3
Keystone Architecture and Trust Model

Keystone: an Open Framework for Architecting Trusted Execution Environments

User (U-mode)
- App
- App

Supervisor (S-mode)
- OS
  - Runtime
  - Enclave
    - App
    - App

Machine (M-mode)
- Security Monitor (SM)
- Trusted Hardware
  - Optional HW
  - Trusted Hardware

Higher Privilege

Trusted

Keystone: an Open Framework for Architecting Trusted Execution Environments

EURO/SYS’20
Keystone Architecture and Trust Model

User (U-mode)

Supervisor (S-mode)

Machine (M-mode)

Security Monitor (SM)

Hardware-Enforced and Software-Defined Isolation
Memory Isolation via RISC-V PMP

Entries

| PMP0 | Accessibility defined by each entry | PMP1 | PMP2 | ... | PMP7 |

| SM   | Enclave 1 | Enclave 2 |

Accessibility
- Can
- Can’t
- Undefined

Higher Priority

Physical Memory
Memory Isolation via RISC-V PMP

Entries | Accessibility defined by each entry
---|---
PMP0 | CAN’T
PMP1 | CAN’t
PMP2 | CAN
... | UNEFined
PMP7 | CAN’T

Physical Memory

Security Monitor (SM)

OS
Runtime
Enclave
App
App
App
App
Keystone Architecture and Trust Model

User (U-mode)

Supervisor (S-mode)

Machine (M-mode)

OS

Runtime

Security Monitor (SM)

What Does Keystone Runtime Do?
What does Keystone Runtime Do?

User (U-mode)
- App
- Enclave App

Supervisor (S-mode)
- OS
- Runtime

Machine (M-mode)
- Security Monitor (SM)
What does Keystone Runtime Do?

User (U-mode)

Supervisor (S-mode)

Machine (M-mode)

Enclave App

seL4 App

Interface

Runtime

Security Monitor (SM)
What does Keystone Runtime Do?

User (U-mode):
- App
- Enclave App
- seL4 App

Supervisor (S-mode):
- OS
- freemem
- paging
- I/O
- syscall
- libc
- seL4

Machine (M-mode):
- Security Monitor (SM)
Memory Management in Keystone

- Enclave self resource management (e.g., dynamic memory resizing)
- Various memory protection mechanisms
Various Memory Protection Mechanisms

- Untrusted
- SM
- Enclave
- Enclave (Encrypted)

LLC
- Baseline
- Cache Partitioning
- On-Chip Enclave
- Software Encryption
- Hardware Encryption

DRAM

Adversary
- SW
- Cache SC
- HW

Keystone: an Open Framework for Architecting Trusted Execution Environments
Various Memory Protection Mechanisms

- Untrusted
- SM
- Enclave
- Enclave (Encrypted)

LLC

DRAM

Baseline

Adversary
- SW
- Cache SC
- HW
Various Memory Protection Mechanisms

Baseline on-
Chip Enclave

Adversary

SW
Cache SC
HW
Various Memory Protection Mechanisms

- LLC
- DRAM

Baseline
Cache Partitioning
On-Chip Enclave

Adversary
SW
Cache SC
HW

Keystone: an Open Framework for Architecting Trusted Execution Environments
Various Memory Protection Mechanisms

Untrusted SM Enclave Enclave (Encrypted)

LLC

DRAM

Baseline

Cache Partitioning

On-Chip Enclave

Software Encryption

Adversary

SW

Cache SC

HW

Keystone: an Open Framework for Architecting Trusted Execution Environments
Various Memory Protection Mechanisms

- Untrusted
- SM
- Enclave
- Enclave (Encrypted)

**LLC**
- Baseline
- Cache Partitioning
- On-Chip Enclave
- Software Encryption
- Hardware Encryption

**DRAM**
- **Adversary**
  - SW
  - Cache SC
  - HW

Keystone: an Open Framework for Architecting Trusted Execution Environments
Evaluation

- Security Analysis
  - Keystone enclave defends various adversary models

- Modularity Analysis
  - Keystone supports fine-grained and modular configuration

- Trusted Computing Base Analysis
  - Various of real-world applications with a few thousands of LoC

- Performance Analysis
  - Security Monitor Overhead
  - Runtime Overhead
  - Cost of Memory Protection Mechanisms
Evaluation

- Security Analysis
  - Keystone enclave defends various adversary models

- Modularity Analysis
  - Keystone supports fine-grained and modular configuration

- Trusted Computing Base Analysis
  - Various of real-world applications with less than thousands of LoC

- Performance Analysis
  - Security Monitor Overhead
  - Runtime Overhead
  - Cost of Memory Protection Mechanisms

Please check our paper!
Runtime Overhead: Memory Management

- Torch benchmark
  - Unmodified NN inference
- Initialization overhead
  - Enclave measurement (SHA3)
- Execution overhead
  - Min -3.12% (LeNet)
  - Max 7.35% (DenseNet)
- Dynamic memory resizing
  - No noticeable overhead
## Cost of Memory Protection Mechanisms

<table>
<thead>
<tr>
<th>Cache Partitioning</th>
<th>On-chip Execution</th>
<th>Software Encryption</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overhead (%)</td>
<td># of Page Faults</td>
<td></td>
</tr>
<tr>
<td>Benchmark</td>
<td></td>
<td></td>
</tr>
<tr>
<td>∅</td>
<td>C</td>
<td>O, P</td>
</tr>
</tbody>
</table>

### Untrusted vs. SM vs. Enclave vs. Enclave (Encrypted)

- **LLC**
  - Baseline
  - Cache Partitioning
  - On-Chip Enclave
  - Software Encryption

- **DRAM**
  - Baseline
  - Cache Partitioning
  - On-Chip Enclave
  - Software Encryption
## Cost of Memory Protection Mechanisms

<table>
<thead>
<tr>
<th>Benchmark</th>
<th>Cache Partitioning</th>
<th>On-chip Execution</th>
<th>Software Encryption</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Overhead (%)</td>
<td></td>
<td># of Page Faults</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>primes</td>
<td>-0.9</td>
<td>40.5</td>
<td>65475.5</td>
</tr>
<tr>
<td>miniz</td>
<td>0.1</td>
<td>128.5</td>
<td>80.2</td>
</tr>
<tr>
<td>aes</td>
<td>-1.1</td>
<td>66.3</td>
<td>1471.0</td>
</tr>
<tr>
<td>bigint</td>
<td>-0.1</td>
<td>1.6</td>
<td>0.4</td>
</tr>
<tr>
<td>qsort</td>
<td>-2.8</td>
<td>-1.3</td>
<td>12446.3</td>
</tr>
<tr>
<td>sha512</td>
<td>-0.1</td>
<td>0.3</td>
<td>-0.1</td>
</tr>
<tr>
<td>norx</td>
<td>0.1</td>
<td>0.9</td>
<td>2590.1</td>
</tr>
<tr>
<td>dhrystone</td>
<td>-0.2</td>
<td>0.3</td>
<td>-0.2</td>
</tr>
</tbody>
</table>
Conclusion

- Introduced Keystone, a customizable framework for TEEs
- Modular design with fine-grained customizability
- Useful for building TEEs for different threat models, functionality, and performance requirements
- Keystone is fully open-source under BSD 3-Clause
  - https://keystone-enclave.org
Thank You!